CS 595: Assignment #5

Due on Thursday, October 16, 2014

Dr Nelson 4:20PM

Victor Nwala

Contents

Problem 1

1. Explore the friendship paradox for your Twitter account. Since Twitter has directional links (i.e., "followers" and "following"), we'll be investigating if the people you follow (Twitter calls these people "friends") follow more people than you. If you are following less than 50 people, use my twitter account "phonedude_mln" instead of your own.

Listing 1: Python script to extract friends of "phonedude_mln" from twitter

```python
import tweepy
import simplejson
from tweepy import OAuthHandler
import itertools
import time

access_token = "384946837-FuQQRJrNFNN8BXyOToG6D1opwRIX1ZQjnysTDjZo"
access_token_secret = "IY1O1Ck8QHZ9rt9X1kkWUhcbRpnfPAN9euOZ3ndc4Gznn"
consumer_key = "mRpX89WMtUSEZSDGi6jtxd5KU"
consumer_secret = "SZyHZiSwQfFMLq5fMwHJTxWqYgfqL9o3AQXdOTP6B4ocQloi9A"

auth = OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)

api = tweepy.API(auth)

user = api.get_user('phonedude_mln')
k = user.screen_name
i = user.friends_count
saveFile = open('friends.txt','a')
saveFile.write(k + ' ' + str(i) )
saveFile.write('\n')
for friend in user.friends(count=i):
    j = friend.screen_name
    print j
    user = api.get_user(j)
    k = user.screen_name
    i = user.friends_count
    saveFile.write(k + ' ' + str(i) )
    saveFile.write('\n')
    #time.sleep(15)
saveFile.close()
```

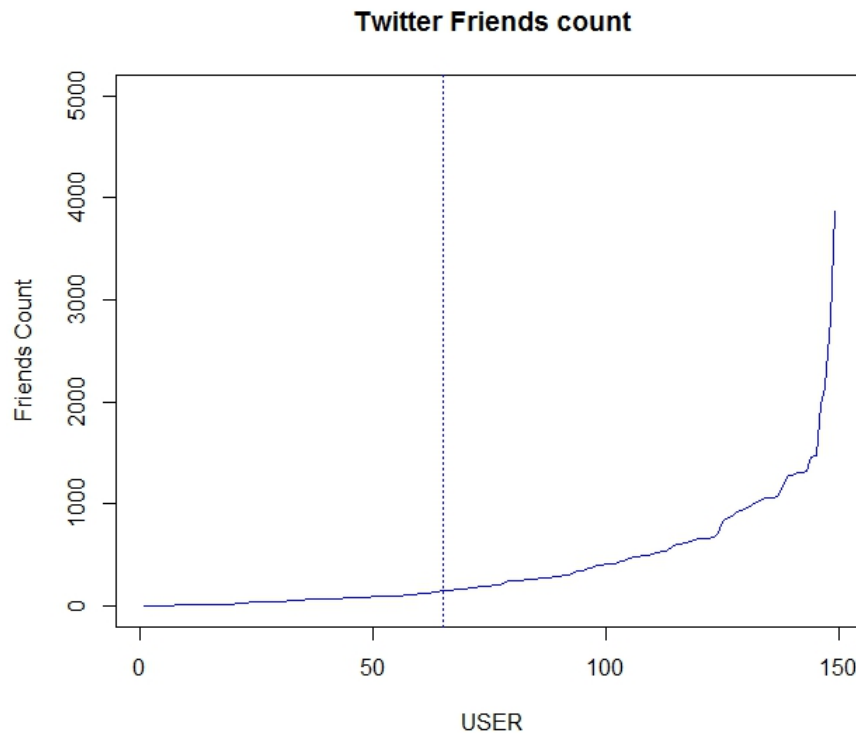The script downloads the friends count of the friends of "phonedude_mln" and store the results in a file.

Figure 1: friendsList.py at work

Figure 2: Twitter Friends Graph

Graph showing the friend count of "phonedude_mln" with vertical line showing his position, 65 on the user axis and a corresponding count of 148 friends

Table 1: Twitter Friends Count Statistics

MEAN	MEDIAN	STANDARD DEVITION
404.7919	191	545.7179

Problem 2

2. Using your facebook account, repeat question #1 (if you have greater than 50 friends).
To answer the question, I used the code credited to "Alexander Nwala" to extract my friends from facebook and their respective friends count.

Listing 2: Python script to extract friends of "Victor Nwala" from Facebook

```
#credit to:
#https://gist.github.com/leostera/3535568
#https://pypi.python.org/pypi/selenium
#cookies problem: http://stackoverflow.com/questions/7854077/using-a-session-cookie-from-selenium-in-urllib2
#http://stackoverflow.com/questions/14583560/selenium-retrieve-data-that-loads-while-scrolling-down

from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time
from selenium.webdriver.common.by import By
import os, sys
from bs4 import BeautifulSoup
import codecs
from random import randint
import getpass
import os
```

```python
     globalHtmlOutputFile = 'allFacebookFriends.html'
20   globalCSVOutputFile = 'facebookFriendFriendsCountTuples.txt'

     #output file: globalHtmlOutputFile
     def getHtmlOfAllFriends(userFaceBookEmail, userFaceBookPassword):

25       if( len(userFaceBookEmail) > 0 and len(userFaceBookPassword) > 0 ):
             pass
         else:
             print "userFaceBookEmail and/or userFaceBookPassword: bad length"
             return
30
         try:
             htmlOutputFile = open(globalHtmlOutputFile, 'w')
         except:
             exc_type, exc_obj, exc_tb = sys.exc_info()
35           fname = os.path.split(exc_tb.tb_frame.f_code.co_filename)[1]
             print fname, exc_tb.tb_lineno, sys.exc_info()
             return

         myFirefoxBrowser = webdriver.Firefox()
40       myFirefoxBrowser.implicitly_wait(3)
         # or you can use Chrome(executable_path="/usr/bin/chromedriver")
         myFirefoxBrowser.get("http://www.facebook.org")
         assert "Facebook" in myFirefoxBrowser.title


45
         elem = myFirefoxBrowser.find_element_by_id("email")
         elem.send_keys(userFaceBookEmail)
         elem = myFirefoxBrowser.find_element_by_id("pass")
         elem.send_keys(userFaceBookPassword)
50       elem.send_keys(Keys.RETURN)


         #http://stackoverflow.com/questions/7854077/using-a-session-cookie-from-selenium-in-urllib2
         all_cookies = myFirefoxBrowser.get_cookies()
55       #cookies = {}
         #for s_cookie in all_cookies:
         #    cookies[s_cookie["name"]]=s_cookie["value"]


60       #open friends page
         friendsLink = 'https://www.facebook.com/friends/'
         myFirefoxBrowser.get(friendsLink)
         myFirefoxBrowser.maximize_window()


65
         #scroll to bottom of page
         previousCountOfFriends = -1
         while True:

70           myFirefoxBrowser.execute_script("return window.scrollTo(0, document.body.scrollHeight);")
             html = myFirefoxBrowser.page_source.encode('utf-8')

             soup = BeautifulSoup(html)
             parentOfUIProfileBlockContent = soup.findAll('div', { 'class' : 'uiProfileBlockContent' })
75
             #lastIndexOfFriends = html.rfind('<div class="uiProfileBlockContent">')
             lastIndexOfFriends = len(parentOfUIProfileBlockContent)

             #'Friends' not found
80           if( lastIndexOfFriends == -1 ):
                 break

             #No new entry
             if( previousCountOfFriends == lastIndexOfFriends ):
85               htmlOutputFile.write(html)
                 break
             else:
                 previousCountOfFriends = lastIndexOfFriends

90           sleepTime = randint(3,7)
             print "...sleeping for", sleepTime, "seconds"
             time.sleep(sleepTime)


95       myFirefoxBrowser.close()
         return previousCountOfFriends

     def getCredentials():

100      userName = ''
         password = ''

         try:
             credentialsFile = open('credentials.txt')
105          credInfo = credentialsFile.readlines()

             if( len(credInfo) > 1 ):
                 userName = credInfo[0]
                 password = credInfo[1]
110      except:
             exc_type, exc_obj, exc_tb = sys.exc_info()
             fname = os.path.split(exc_tb.tb_frame.f_code.co_filename)[1]
             print fname, exc_tb.tb_lineno, sys.exc_info()
             return
```

```python
115
        return userName, password

    def getFBHtmlDump(inputFileName):
120
        htmlText = ''

        if( len(inputFileName) > 0 ):
            try:
125             inputFile = open(inputFileName, 'r')
                htmlText = inputFile.read()
            except:
                exc_type, exc_obj, exc_tb = sys.exc_info()
                fname = os.path.split(exc_tb.tb_frame.f_code.co_filename)[1]
130             print fname, exc_tb.tb_lineno, sys.exc_info()


        return htmlText

135 #writes tuples <friend, friendCount> into globalCSVOutputFile
    def getFriendOfFriendsFromHtml(htmlText):

        goAheadFlag = False

140     if( len(htmlText) > 0 ):

            try:
                outputFile = codecs.open(globalCSVOutputFile, 'w', 'utf-8')
                outputFile.write('"USER", "FRIENDCOUNT"\n')
145         except:
                exc_type, exc_obj, exc_tb = sys.exc_info()
                fname = os.path.split(exc_tb.tb_frame.f_code.co_filename)[1]
                print fname, exc_tb.tb_lineno, sys.exc_info()
                return
150

            soup = BeautifulSoup(htmlText)
            parentOfUIProfileBlockContent = soup.findAll('div', { 'class' : 'uiProfileBlockContent' })

155         for profile in parentOfUIProfileBlockContent:

                    friendName = profile.find('div', { 'class' : 'fsl fwb fcb' })
                    potentialFriendsCount = profile.find('a', { 'class' : 'uiLinkSubtle' })

160                 #potentialFriendsCount: x (f)riends | x mutual friends, etc, so split
                    if( potentialFriendsCount is not None ):

                        potentialFriendsCount = potentialFriendsCount.text.split(' ')

165                     if( len(potentialFriendsCount) > 1 ):
                            if( len(potentialFriendsCount[1]) > 0):
                                if( potentialFriendsCount[1][0].lower() == 'f' ):

                                    friendCount = potentialFriendsCount[0].replace(',','')
170
                                    stringToWrite = friendName.text + ', ' + friendCount + '\n'
                                    outputFile.write(stringToWrite)
                                    goAheadFlag = True

175
            outputFile.close()


        return goAheadFlag
180
    if __name__ == "__main__":


        print ''
185
        print 'Welcome to get fb friends of friends. If all goes well,'
        print 'The application will write your fb friends of friends into ./' + globalCSVOutputFile

        print ''
190     userNameFacebook = raw_input("Email ID: ")
        passwordFacebook = getpass.getpass('Password: ')

        userNameFacebook = str(userNameFacebook)
        passwordFacebook = str(passwordFacebook)
195
        userNameFacebook = userNameFacebook.strip()
        passwordFacebook = passwordFacebook.strip()

        intGoAheadFlag = getHtmlOfAllFriends(userNameFacebook, passwordFacebook)
200
        if ( intGoAheadFlag > -1 ):
            facebookDumpInputFileName = globalHtmlOutputFile
            htmlText = getFBHtmlDump(facebookDumpInputFileName)
            boolGoAhead = getFriendOfFriendsFromHtml(htmlText)
205
            #open file
            if( boolGoAhead ):
                myFirefoxBrowser = webdriver.Firefox()
                filePath = 'file:///' + os.getcwd() + '/' + globalCSVOutputFile
210             myFirefoxBrowser.get(filePath)
```
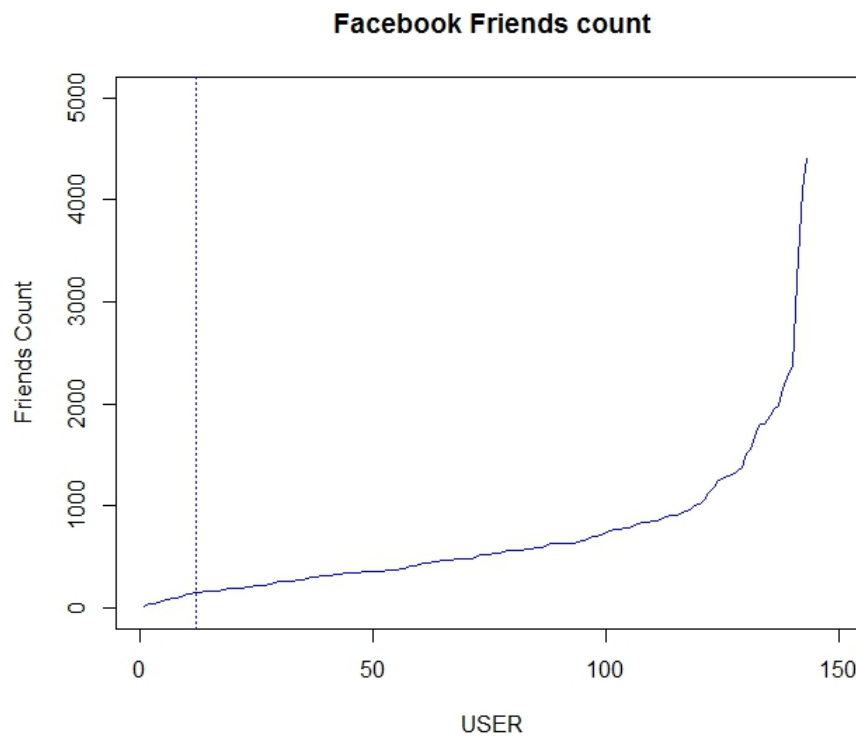
Figure 3: seleniumScrapePB.py at work

Figure 4: Facebook Friends Graph

Graph showing the friend count of "Victor Nwala" with vertical line showing his position, 12 on the user axis and a corresponding count of 155 friends

Table 2: Facebook Friends Count Statistics

MEAN	MEDIAN	STANDARD DEVITION
691.2238	490	681.5631

In conclusion, the font size in my code listing is smaller than usual so that my code will all fit into the defined text area. Secondly, the facebook extraction code, seleniumScrapePB.py extracts only friend counts of friends whose counts are visible. Finally, the mean, median and standard deviation calculations where all done in Rstudio with a tool called, favstats. I also used the "abline()" function to specify the exact locations for my vertical line on both graphs in Rstudio. I should also state that the "Tweepy Documentation" was very helpful for problem 1.